**JP MORGAN SOFTWARE ENGINEERING VIRTUAL INTERNSHIP TASKS :**

**Task1 : Interface with a stock price data feed**

### **Here is the background information on your task**

You’ve been asked to assist with some development to add a chart to a trader’s dashboard allowing them to better identify under/over-valued stocks.

The trader would like to be able to monitor two historically correlated stocks and be able to visualize when the correlation between the two weakens (i.e. one stock moves proportionally more than the historical correlation would imply). This could indicate a potential trade strategy to simultaneously buy the relatively underperforming stock and sell the relatively outperforming stock. Assuming the two prices subsequently converge, the trade should be profitable.

Most data visualization for our traders is built on JPMorgan Chase's Perspective data visualization software, which is now open source. If you want to explore that, a link is provided in the resources section.

Before implementing this request using perspective, first you’ll need to interface with the relevant financial data feed and make the necessary adjustments to facilitate the monitoring of potential trade opportunities.

\* Understanding the finance and trading part is not required.

\* Being familiar with python scripting language and command line basics is not required as you will be guided in this exercise

\* (Note, you DO NOT have to install Perspective as an individual software onto your machine. All you need to complete this task is to follow the instructions in step 3)

**Here is your task**

**For the first module of this project will need you to accomplish the following:**

1. Set up your system by downloading the necessary repository, files, tools and dependencies
2. Fix the broken client datafeed script in the repository by making the required adjustments to it.
3. Generate a patch file of the changes you made
4. Bonus task: Add unit tests in the test script in the repository.

**We've broken down the steps for you in stages below so you can accomplish this task in an organized manner.**

**Set Up**

Before you can tackle any software or development task you need to set up your development environment. Your development environment refers to your system having all the required software installed to modify the code, as well as getting the code of the project itself onto your computer.

**Task2 : Use Jp Morgan Chase Fremework and tools**

### **Here is the background information on your task**

Typically, traders monitor stock prices and trading strategies by having data displayed visually on their screens in chart form. Often these charts will be accompanied by alerts that notify users when certain events occur or when preset price thresholds are hit.

JPMorgan Chase created the Perspective tool over many years to allows users to present and manipulate data feeds visually in web applications.

Perspective provides a set of flexible data transforms, such as pivots, filters, and aggregations. It utilizes bleeding-edge browser technology such as Web Assembly and Apache Arrow and is unmatched in browser performance. It is engineered for reliability and production-vetted on the JPMorgan Chase trading floor and is now available to the development community as Open Source. If you want to explore that, a link is provided in the resources section.

\* Understanding the finance and trading part is not required for this task.

\* Being familiar with python scripting language, command line basics, javascript, react and typescript are not required for this task too as you will be guided in this exercise

\* (Note, you DO NOT have to install Perspective as an individual software onto your machine. All you need to complete this task is to follow the instructions in step 3)**Here is your task**

**For the second module of this project will need you to accomplish the following:**

1. Set up your system by downloading the necessary files, tools and dependencies.
2. Fix the broken typescript files in repository to make the web application output correctly
3. Generate a patch file of the changes you made.

**We've broken down the steps for you in stages below so you can accomplish this task in an organized manner.**

**Set Up**

Before you can tackle any software or development task you need to set up your development environment. Your development environment refers to your system having all the required software installed to modify the code, as well as getting the code of the project itself onto your computer.

**Making Changes**  
When you’re in a work environment, you’ll usually receive tasks in the form of engineering tickets.  
Here is an example of what this task looks like in the form of an engineering ticket.  
  
Purpose:  
The objective of this task will be for you to fix the client-side web application so that it displays a graph that automatically updates as it gets data from the server application (*see Before and After images below*) Currently, the web application only gets data every time you click on the 'Start Streaming Data' button and does not aggregate duplicated data.  
  
Acceptance Criteria:

* This ticket is done when the graph displayed in the client-side web application is a continuously updating line graph whose y axis is the stock’s top\_ask\_price and the x-axis is the timestamp of the stock. The continuous updates to the graph should be the result of continuous requests and responses to and from the server for the stock data.
* This ticket is done when the graph is also able to aggregate duplicated data retrieved from the server

To properly make the changes necessary to complete the objectives of this task, follow the guide below.

**Task3 : Display data visually for traders**

### **Here is the background information on your task**

Being able to access and  adjust data feeds is critical to any trading analysis and stock price monitoring. From the previous tasks, we now have the adjusted data set up on your systems and being piped into Perspective.

For traders to have a complete picture of all the trading strategies being monitored, several screens typically display an assortment of live and historical data at their workstation.

Given there is a lot of information and data being produced at once, visualizing data in a clear manner with UI/UX considerations accounted for is critical to providing traders with the tools to improve their performance.

\* Understanding the finance and trading part is not required for this task.

\* Being familiar with python scripting language, command line basics, javascript, react and typescript are not required for this task too as you will be guided in this exercise

\* (Note, you DO NOT have to install Perspective as an individual software onto your machine. All you need to complete this task is to follow the instructions in step 3)

**Here is your task**

**For the third module of this project will need you to accomplish the following:**

1. Set up your system by downloading the necessary files, tools and dependencies.
2. Modify the typescript files in repository to make the web application behave in the expected manner
3. Generate a patch file of the changes you made.

**We've broken down the steps for you in stages below so you can accomplish this task in an organized manner.**

**Set Up**

Before you can tackle any software or development task you need to set up your development environment. Your development environment refers to your system having all the required software installed to modify the code, as well as getting the code of the project itself onto your computer.

To do this we've created a simple PDF guide below on how to get your environment set up: